# Characterizing the Adversarial Power in Uniform and Ergodic Node Sampling

Emmanuelle Anceaume
IRISA / CNRS
Campus Universitaire
de Beaulieu – F-35042
Rennes, France
emmanuelle.anceaume@irisa.fr

Yann Busnel
LINA
Université de Nantes
2, rue de la Houssinière
BP 92208 – F-44322
Nantes, France
Yann.Busnel@univ-nantes.fr

Sébastien Gambs
IRISA / Université de
Rennes 1 — INRIA
Campus Universitaire
de Beaulieu – F-35042
Rennes, France
sebastien.gambs@irisa.fr

## ABSTRACT

In this paper, we consider the problem of achieving uniform and ergodic peer sampling in large scale dynamic systems under adversarial behaviors. The main challenge is to guarantee that any honest node is able to construct a uniform and non-fixed (ergodic) sample of the node identifiers in the system, and this, despite the presence of malicious nodes controlled by an adversary. This sample is built out of a stream of events received at each node. We consider and study two types of adversary: an omniscient adversary that has the capacity to eavesdrop all the messages that are exchanged within the system, and a blind adversary that can only observe messages that have been sent or received by the manipulated nodes. The former model allows us to derive lower bounds on the impact that the adversary has on the sampling functionality while the latter one corresponds to a realistic model. Given any sampling strategy, we quantify the minimum effort exerted by both types of adversary on any input stream to prevent this strategy from outputting a uniform and ergodic sample.

## Categories and Subject Descriptors

F.2.2 [**Theory of Computation**]: Analysis of Algorithms and Problem Complexity—*Non-numerical Algorithms and Problems*

## General Terms

Theory, Algorithms

## 1. INTRODUCTION

We investigate the problem of achieving uniform and ergodic peer sampling in large scale open systems in presence of adversarial nodes. Uniform sampling is a fundamental primitive ensuring that any individual in a population has the same probability to be selected as sample. Uniform sampling finds its root in many practical problems such as data collection, data dissemination, event driven communication, load balancing and data-caching [6, 12, 14, 17]. In particular, publish-subscribe systems require that each subscriber of any given topic has the same probability to be chosen as access point for that topic, even in case some topics are more popular than others [5]. Having access to a peer sampling service provides such systems with uniform samples of nodes. In the context of event-driven interaction systems, population protocols rely on uniform node sampling as a basis for the construction of fair schedulers [6]. Achieving uniform sampling in large scale open and dynamic systems has been shown to be difficult. One of the reasons is that the population of these systems is very large (*e.g.*, thousands or millions of nodes) and exhibits a very high churn (recent studies on the eDonkey file-sharing network have shown that in average $500,000$ peers connect and disconnect per day [15]). Moreover, openness makes unavoidable the presence of malicious nodes that try to subvert the system functionalities. Most common attacks mainly consist in isolating targeted honest nodes from the remaining of the system so that sought resources and/or services become unreachable (denial of services attacks), and thus to the eventual partitioning of the system.

By relying on the topological properties of structured large scale dynamic systems, it has been shown that it is possible to guarantee that with high probability the identifier of any node is equally likely to appear in the local view of each other honest node in a number of communication rounds polynomial in the size of the system. One way to achieve this is by imposing nodes to frequently depart from their position and move to another random position in the system [2, 4]. In unstructured large scale systems, nodes cannot rely on the topological nature of structured graphs to detect undesirable behaviors. To circumvent this issue, Bortnikov *et al.* [7] rely on the properties of min-wise independent permutations. Such permutations are fed by the streams of gossiped nodes identifiers and eventually converge towards a uniform sampling on the node identifiers. However, this sample is definitive in the sense that no other node identifier received in the future in the input stream will ever have the chance to appear in the random sample. This makes this sampling strategy uniform but not *ergodic*, that is, it does not guarantee that the identifier of any node in the system


*The 1st International Workshop on Algorithms and Models for Distributed Event Processing (AlMoDEP '11)* collocated with the 25th International Symposium on Distributed Computing (DISC 2011). September 19, 2011 – Rome, Italy

has a non-zero probability to appear infinitely often in a sample. This clearly makes such an approach unfitted for dynamic systems.

A preliminary step in determining conditions under which uniform and ergodic sampling is achievable in unstructured peer-to-peer systems potentially populated with a large proportion of malicious nodes has been presented in a previous paper [3]. Briefly, we show in [3] that imposing strict restrictions on the number of messages sent by malicious nodes during a given period of time and providing each honest node with a very large memory (proportional to the size of the system) is a necessary and sufficient condition to obtain uniform and ergodic sampling.

In the present paper, we propose a characterization of the adversarial power towards biasing uniform and ergodic sampling. By adopting a statistical view of the input stream and by comparing distributions using metrics such as information divergence, we derive lower bounds on the work that the adversary has to exert to bias this input stream so that uniform and ergodic sampling does not hold. We consider and study two models of adversary: the omniscient adversary, which has the capacity to eavesdrop on all the messages that are exchanged within the system, and the blind adversary, which can only observe messages that have been sent or received by malicious nodes. To the best of our knowledge, we are not aware of any previous work that has characterized the minimum effort an adversary has to exert to prevent the uniform and ergodic sampling to be achievable.

The outline of this paper is as follows. In Section 2, we give an overview of the existing related work. Section 3 describes the model of the system and the assumptions that are made. In Section 4, we describe the functionalities of a sampling component and the properties that it should guarantee while in Section 5, we present some background on information divergence of data streams. The omniscient and blind adversary models, as well as the characterization of the minimum effort the adversary has to exert to bias the sampling properties, are respectively studied in Sections 6 and 7. Finally, Section 8 concludes with some open issues.

## 2. RELATED WORK

Different approaches have been proposed to deal with malicious behaviors in the peer sampling problem in unstructured large scale dynamic systems. Jesi *et al.* [13] propose a random sampling algorithm taking explicitly into account malicious nodes. Their solution assumes that the ultimate goal of the malicious nodes is to mutate the random graph into a hub-based graph, hub for which malicious nodes gain the lead. This approach, also adopted in several structured based systems [21] through auditing mechanisms, or in sensor networks [16], is effective only if the number of malicious nodes is very small with respect to the size of the system (*i.e.*, typically of $O(\log n)$). Bortnikov *et al.* [7] have recently proposed a uniform but non-ergodic peer sampling algorithm that tolerates up to a linear number of malicious nodes. Their sampling mechanism exploits the properties offered by min-wise permutations. Specifically, the sampling component is fed with the stream of node identifiers periodically gossiped by nodes, and outputs the node identifier whose image value under the randomly chosen permutation is the smallest value ever encountered. Thus eventually, by the property of min-wise permutation, the sampler converges towards a random but permanent sample. In a previous work [3], the authors show that imposing strict restrictions on the number of messages sent by malicious nodes during a given period of time and providing each honest node with a very large memory (proportional to the size of the system) are necessary and sufficient conditions to obtain uniform and ergodic (non permanent) sampling. It is worth noting that our results complement two previous results [8, 11], in which an analysis of the class of uniform and ergodic sampling protocols is presented. Both previous work provide a complete analytical proof of a gossip-based protocol that reaches both uniformity and ergodicity, but in contrast to the present work, adversarial behaviors were not considered. Finally, taking a completely different approach from the previously mentioned papers, which are based on gossip algorithms or on distance function properties, the techniques presented in [22, 23] rely on social network topologies to guard against Sybil attacks. Both protocols take advantage of the fact that Sybil attacks try to alter the fast mixing property of social networks to defend against these attacks.

## 3. SYSTEM MODEL

**Model of the Network.** We consider a dynamic system populated by a large collection of nodes in which each node is assigned a unique and permanent random identifier from an $m$-bit identifier space. Node identifiers (simply denoted *ids* in the following) are derived by applying some standard strong cryptographic hash function on nodes intrinsic characteristics. The value of $m$ (160 for the standard SHA-1 hash function) is chosen to be large enough to make the probability of identifiers collision negligible. The system is subject to churn, which is classically defined as the rate of turnover of nodes in the system [10]. Each node knows only a small set of nodes existing within the system and this knowledge generally varies according to the activity of the system. The particular algorithm used by nodes to update this small set and to route messages induces the resulting overlay topology. In this work, we consider only unstructured overlays. Unstructured overlays are assumed to conform with random graphs, in the sense that relationships among nodes are mostly set according to a random process.

**Adversary.** We assume the presence of malicious nodes that try to manipulate the system by exhibiting undesirable behaviors (a node that is not malicious is called honest). In our context, this amounts to dropping messages that should normally be relayed by malicious nodes towards honest ones, and injecting new messages. Injecting new messages does not mean that malicious nodes have the ability to impersonate honest nodes. Rather, their goal is to judiciously increase the frequency of chosen ids to bias the sample list maintained by nodes. We model malicious behaviors through an adversary that fully controls these malicious nodes. The adversary model we considered follows the lines of [7, 13, 18], however we distinguish between two types of adversary: the *omniscient* adversary that is able to eavesdrop all messages exchanged within the system, and the *blind* adversary that can only observe messages sent or received by malicious nodes. In both models, we assume that the adversary can neither drop a message exchanged between two honest nodes nor tamper with its content without being detected. This is achieved by assuming the existence of a signature scheme (and the corresponding public-key infrastructure) ensuring the authenticity and integrity of messages.

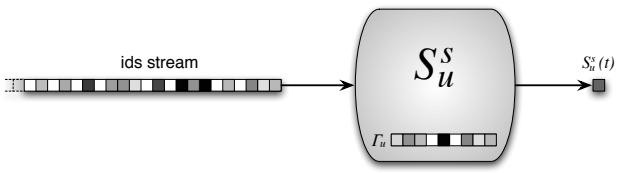


**Figure 1: Sampling component of node $u \in \mathcal{N}$.**

**Sampling Assumptions.** Similarly to Bortnikov *et al.* [7], we assume that there exists a time $T_0$ such that after that time, the churn of the system ceases. This assumption is necessary to make the notion of uniform sample meaningful. Thus from $T_0$ onwards, the population of the system $\mathcal{S}$ is composed of $n \ll 2^m$ nodes, such that at least $(1-k)n$ of them are honest and no more than $kn$ of them are malicious, and thus are controlled by the adversary. The subset of honest nodes in the overlay is denoted by $\mathcal{N}$. Finally, we assume that all the nodes in $\mathcal{S}$ are weakly connected from time $T_0$ onwards, which means that there exists a path between any pair of nodes in $\mathcal{S}$ in the underlying undirected graph whose vertices represent the nodes of $\mathcal{S}$ and edges are the communication links between these nodes.

# 4. SAMPLING COMPONENT

Following the approach taken in [3], each node $u \in \mathcal{N}$ has locally access to a *sampling component*[1] as presented in Figure 1. The sampling component implements a *strategy* $s$ and has uniquely access to a data structure $\Gamma_u$, referred to as the *sampling memory*. The size of the sampling memory $\Gamma_u$ is bounded and is denoted by $|\Gamma_u|$. The sampling component $S_u^s$ is fed with an infinite stream $< v_i, v_j, \ldots >$ of (possibly non unique) node ids that correspond to the node ids periodically received by node $u \in \mathcal{N}$. This stream results either from the propagation of node ids through gossip-based algorithms (namely through push, or pull or push-pull mechanisms initiated by $u$ and its neighbors), or from the node ids received during random walks initiated at $u$, or even from induced churn. The fingerprint of an input stream is a collection of weighted points in which each node id is weighted by the number of times this node id appears in the stream. Specifically, a stream of node ids can be summarized by $< (v_1, m_1), \ldots, (v_n, m_n) >$, where $v_i$ denotes the identifier of a node in $\mathcal{S}$ and $m_i \in \mathbb{N}$ represents the number of times $v_i$ appears in the stream. At each time $t$, the following three steps are atomically executed: the first element of the stream, say node id $v$, is given as input to the sampling component. The sampling component $S_u^s$ reads $v$, and removes it from the stream. According to its strategy $s$, $S_u^s$ may store or not $v$ in $\Gamma_u$ and outputs at most one node id.

For example, the strategy $s$ may consist in storing $v$ if $\Gamma_u$ is not full, or in substituting $v$ for a randomly chosen node id that belongs to $\Gamma_u$, or in simply dropping $v$. The output at time $t$, denoted $S_u^s(t)$, is chosen among the node ids in $\Gamma_u$ according to strategy $s$. For instance, strategy $s$ may consist in choosing the smallest node id in $\Gamma_u$, or the smallest node id under a given min-wise permutation [7]. The maximum finite hitting time needed for the sampling component $S_u^s$ to reach a uniform sample is denoted by $T_s$. Clearly, $T_s$ depends on the strategy $s$ implemented by the sampling component and also on the stream of node ids the sampling component is fed with. We assume that the sampling strategy is known by the adversary in the sense that the algorithm used is public knowledge. However, if the algorithm is a randomized one, the adversary does not have access to the local random coins used by the honest nodes.

Finally, $\delta$ represents the number of node ids injected by the adversary in the input stream of node $u$ during the time interval $T_s$. Note that it does not matter whether the injected node ids correspond to the node ids of malicious nodes or not as the unique goal of the adversary is to bias the input stream in such a way that whatever the strategy $s$ of the sampler component, its output $S_u^s(t)$ cannot guarantee both the uniform and ergodic properties [3]. More precisely, these properties are defined as follows

PROPERTY 4.1 (UNIFORMITY). *Let $\mathcal{N}$ be a weakly connected graph from time $T_0$ onwards, then for any time $t \geq T_s$, for any node $v \in \mathcal{S}$, and for any node $u \in \mathcal{N}$,*

$$\Pr[v \in S_u^s(t)] = \frac{1}{|\mathcal{S}|}.$$

PROPERTY 4.2 (ERGODICITY). *Let $\mathcal{N}$ be a weakly connected graph from time $T_0$ onwards, then for any time $t \geq T_s$, for any node $v \in \mathcal{S}$, and for any node $u \in \mathcal{N}$,*

$$\Pr\left[\{t' | t' > t \wedge v \in S_u^s(t')\} = \emptyset\right] = 0,$$

*where $\emptyset$ represents the empty set.*

Uniformity states that for any node in the system, its node id should have the same probability to appear in the sample of honest nodes in the system, while ergodicity says that any node id should have a non-zero probability to appear infinitely often in the sample of each honest node in the system. Note that uniformity by itself does not imply ergodicity, and *vice versa*. Indeed, the former does not impose any restriction on the freshness of output node ids, while the latter one does not provide any guarantee regarding the equiprobability of node ids to be chosen as samples. Moreover, as each node $v$ in $\mathcal{S}$ has a non-zero probability to be returned by $S_u^s(t)$ at time $t$, $v$ must appear at least once in the input stream. Thus, $\forall v \in \mathcal{S}$, starting from time $T_s$, $m_v > 0$. Note that as previously said the model and analysis presented in this paper are independent from the way the stream of node ids at each node $u$ is generated.

# 5. INFORMATION DIVERGENCE OF DATA STREAMS

A natural approach to detect changes on data streams is to model it as a distribution and to compute the distance between the observed stream and the ideal one. The metric we use in our context is the Kullback-Leibler (KL) divergence. It is sometimes called the relative entropy [9].

DEFINITION 5.1 (KULLBACK-LEIBLER DIVERGENCE). *Given two probability distributions on events $p = \{p_1, \ldots, p_n\}$ and $q = \{q_1, \ldots, q_n\}$, the Kullback-Leibler divergence between $p_i$ relative to $q_i$ is defined as the expected value of the likelihood ratio with respect to $q_i$. Specifically,*

$$\mathcal{D}(p||q) = \sum_{i=1}^{n} p_i \log_2 \frac{p_i}{q_i} = H(p,q) - H(p), \quad (1)$$

[1]Although malicious nodes have also access to a sampling component, we cannot impose any assumptions on how they feed it or use it as their behavior can be totally arbitrary.

*where $H(p) = -\sum p_i \log_2 p_i$ is the (Shannon) entropy of $p$ and $H(p,q) = -\sum p_i \log_2 q_i$ is the cross entropy of $p$ and $q$ (by convention, $0 \log_2 0 = 0$).*

The KL-divergence is a member of a larger class of distances known as the Ali-Silvey distances [1]. For the sake of clarity, we will use the notation log to denote the logarithm in base 2. Let $p^{(\mathcal{U})}$ be the uniform distribution corresponding to a uniform stream, *i.e.*, $\forall i \in [1..n], p_i^{(\mathcal{U})} = \frac{1}{n}$, and $q$ be the probability distribution corresponding to the input stream. In the rest of this paper and according to the classical use of the KL-divergence, we consider $\mathcal{D}(q||p^{(\mathcal{U})})$ as a measure of the divergence of the stream from the ideal one.

DEFINITION 5.2 ($\tau$-CLOSENESS). *A stream of node ids $\sigma$ is $\tau$-close if the KL-divergence between the probability distribution $q$ corresponding to $\sigma$ and the uniform probability distribution $p^{(\mathcal{U})}$ is below or equal to a given value $\tau$, where $\tau$ is a real. In the sequel, $\tau$ is called the robustness threshold.*

Finally, given two distributions of probability, the Earth Mover's Distance (EMD) [20] measures the minimal amount of work needed to transform one distribution to another by moving the probability mass between events. We rely on this metric to quantify the effort that an adversary exerts to bias the input stream. In our context, a unit of work corresponds to dropping one id and to pushing another id instead in the input stream.

# 6. OMNISCIENT ADVERSARY MODEL

In this section, we study the behavior of an omniscient adversary, which has the capacity to eavesdrop on all the messages sent and received by all the nodes in $\mathcal{S}$. We demonstrate that the strategy that pushes all the probability mass over a single id is the one that maximizes the bias of the input stream so that it becomes far from the uniform distribution. We also describe an optimal strategy on how to achieve it.

In the following, the analysis of infinite input stream is restricted to any window of length $T_s$ observed from time $t \geq T_0$. As previously said, $T_s$ depends on the sampler strategy, and thus can be arbitrarily large. By abuse of language, the term "stream" will thus denote in the remaining of the paper the stream restricted to any such window of length $T_s$. The notation $[1..n]$ denotes the set $\{1, 2, \ldots, n\}$.

Let $\overline{\sigma}$ be a stream such that the id of each node in $\mathcal{S}$ appears exactly once in the stream, except for a unique id that appears in all the remaining slots. Therefore, there exists a unique $v_i \in \mathcal{S}$ such that $m_{v_i} = T_s - (n-1)$ and $\forall v_j \neq v_i \in \mathcal{S}, m_{v_j} = 1$. The following theorem states that the probability distribution associated to this particular stream is the one that has the maximal divergence from the uniform distribution.

THEOREM 6.1. (MAXIMAL DIVERGENCE FROM THE UNIFORM DISTRIBUTION)
*Let $p^{(\mathcal{U})}$ be the uniform distribution corresponding to a uniform stream, that is, $\forall i \in [1..n], p_i^{(\mathcal{U})} = \frac{1}{n}$, and $\overline{q}$ be the probability distribution corresponding to $\overline{\sigma}$,* i.e.*, it exists a unique $v_i \in \mathcal{S}, \overline{q}_{v_i} = \frac{T_s-(n-1)}{T_s}$ and $\forall v_j \in \mathcal{S}, v_j \neq v_i \Rightarrow \overline{q}_{v_j} = \frac{1}{T_s}$. Then, for any possible probability distribution $q$,*

$$\mathcal{D}(q||p^{(\mathcal{U})}) \leq \mathcal{D}(\overline{q}||p^{(\mathcal{U})}).$$

PROOF. Let $q$ be the probability distribution representing any valid input stream on $(T_0, T_s]$. We have $\forall v_i \in \mathcal{S}, q_{v_i} = \frac{m_{v_i}}{T_s}$, where $m_{v_i}$ is the number of times $v_i$ is present in the input stream. We have

$$\begin{aligned}\mathcal{D}(q||p^{(\mathcal{U})}) &= H(q, p^{(\mathcal{U})}) - H(q) \\ &= -\sum_{i=1}^{n} q_i \log\left(p_i^{(\mathcal{U})}\right) - H(q) = \log(n) - H(q).\end{aligned}$$

Therefore, maximizing $D(q||p^{(\mathcal{U})})$ amounts to minimizing $H(q)$, which is equivalent to maximize $\sum_{i=1}^{n} m_{v_i} \log\left(\frac{m_{v_i}}{T_s}\right)$. We characterize the stream that minimizes $H(q)$ under the following constraints:

$$\begin{cases} 1 \leq m_{v_i} \leq T_s \quad \text{with } 1 \leq i \leq n, \\ \sum_{i=1}^{n} m_{v_i} = T_s. \end{cases} \tag{2}$$

From this set of constraints, we immediately have $1 \leq m_{v_i} \leq T_s - (n-1)$. To relax the second constraint, we consider $m_{v_n} = T_s - \sum_{i=1}^{n-1} m_{v_i}$. Let function $f$ be such that

$$\begin{aligned}f(m_{v_1}, \ldots, m_{v_{n-1}}) = &\sum_{i=1}^{n-1} m_{v_i} \log\left(\frac{m_{v_i}}{T_s}\right) \\ &+ \left(T_s - \sum_{i=1}^{n-1} m_{v_i}\right) \log\left(1 - \sum_{i=1}^{n-1} \frac{m_{v_i}}{T_s}\right).\end{aligned}$$

Function $f$ is differentiable on its domain $\mathcal{I}_s = [1..T_s - n + 1]^{n-1}$, thus we get

$$\begin{aligned}\frac{\mathrm{d}f}{\mathrm{d}m_{v_j}}(m_{v_1}, \ldots, m_{v_{n-1}}) &= \log\left(\frac{m_{v_j}}{T_s}\right) + m_{v_j}\frac{T_s}{m_{v_j}} \\ &+ \log\left(1 - \sum_{i=1}^{n-1} \frac{m_{v_i}}{T_s}\right) + \frac{T_s - \sum_{i=1}^{n-1} m_{v_i}}{1 - \sum_{i=1}^{n-1} \frac{m_{v_i}}{T_s}} \\ &= \log\left(m_{v_j}\right) + \log\left(T_s - \sum_{i=1}^{n-1} m_{v_j}\right) + 2(T_s - \log(T_s)).\end{aligned}$$

According to Equation 2, we have

$$\log\left(m_{v_j}\right) + \log\left(T_s - \sum_{i=1}^{n-1} m_{v_j}\right) \geq 0$$

and, as $T_s \gg 1$, this implies $T_s - \log(T_s) > 0$. Then, we obtain that $\frac{\mathrm{d}f}{\mathrm{d}m_{v_j}} > 0$, leading to the fact that $f$ is strictly increasing according to $m_{v_j}$. The maximum is then reach for $m_{v_j} = T_s - n + 1$ ($f$ is a Schur-convex function).

From the set of constraints (*cf.* Equation 2), if the maximum of $\mathcal{D}(q||p^{(\mathcal{U})})$ is reached for $m_{v_j} = T_s - n + 1$ then $\sum_{i=1, i\neq j}^{n} m_{v_i} = n-1$ implies that $\forall i \in [1..n], i \neq j, m_{v_i} = 1$, which concludes the proof. □

This allows us to formulate an upper-bound $\mathcal{D}^{\max}$ on the KL-divergence between the uniform stream and any other stream:

$$\begin{aligned}\mathcal{D}^{\max} &= \mathcal{D}(\overline{q}||p^{(\mathcal{U})}) \\ &= \log(n) + \log(T_s) - \left(1 - \frac{n-1}{T_s}\right)\log\left(T_s - n + 1\right).\end{aligned} \tag{3}$$

Thus any input stream $\sigma$ is $\mathcal{D}^{\max}$-close (*cf.* Definition 5.2).

To determine the minimal effort that the adversary has to exert to bias the input stream so that both uniformity and ergodicity properties do not hold, we use the Earth Mover's Distance (EMD) between the uniform distribution and the target one. In the following, when we say that the adversary replaces node id $v_i$ by node id $v_j$, we mean that he drops $v_i$ from the input stream and injects $v_j$ instead. Recall that the adversary is able to drop and inject node ids only from the nodes it controls. However, as motivated in the introduction, the adversary may succeed in surrounding a honest node with malicious nodes so that it may shape by itself the input stream of this honest node.

LEMMA 6.2. (OPTIMAL STRATEGY TO MAXIMIZE THE DIVERGENCE)
*Given an input stream $\sigma$, replacing the less frequent node id in $\sigma$ with the most frequent one maximizes the gain in KL-divergence with respect to the uniform distribution for the same amount of work as measured by the EMD distance.*

PROOF. Given an input stream $\sigma$ represented by the probability distribution $q$, we construct the input stream $\sigma'$ from $\sigma$ by substituting one occurrence of node id $v_i$ with node id $v_j$ so that $\mathcal{D}(q'||p^{(\mathcal{U})})$ is maximized after this replacement (where $q'$ denote the probability distribution representing $\sigma'$). This amounts to maximizing $\left[\mathcal{D}(q'||p^{(\mathcal{U})}) - \mathcal{D}(q||p^{(\mathcal{U})})\right]$. Recall that all node ids in $\mathcal{S}$ must be present in $\sigma'$. Therefore, we search for the node id pair $(v_i, v_j)$ such that

$$\begin{cases} m'_{v_j} &= m_{v_j} + 1 \\ m'_{v_i} &= m_{v_i} - 1 \\ v_j &= \arg\max_{v_j \in \mathcal{S}} \left(q'_{v_j} \log\left(q'_{v_j}\right) - q_{v_j} \log\left(q_{v_j}\right)\right) \\ v_i &= \arg\max_{v_i \in \mathcal{S}} \left(q'_{v_i} \log\left(q'_{v_i}\right) - q_{v_i} \log\left(q_{v_i}\right)\right) \end{cases}$$

Consider the function $f : x \mapsto x\log(x)$, which is strictly increasing. As for any $k \in \mathcal{S}$, $q_{v_k} = m_{v_k}/T_s$, we have:

$$\begin{cases} v_j = \arg\max_{v_j \in \mathcal{S}} \left(f\left(\frac{m_{v_j}+1}{T_s}\right) - f\left(\frac{m_{v_j}}{T_s}\right)\right) \\ v_i = \arg\max_{v_i \in \mathcal{S}} \left(f\left(\frac{m_{v_i}-1}{T_s}\right) - f\left(\frac{m_{v_i}}{T_s}\right)\right) \end{cases}$$

$$\Longrightarrow \begin{cases} v_j = \arg\max_{v_j \in \mathcal{S}} \ m_{v_j} \\ v_i = \arg\min_{v_i \in \mathcal{S}} \ m_{v_i} \end{cases}$$

This leads from the fact that the function $g_1 : x \mapsto f(\frac{x+1}{T_s}) - f(\frac{x}{T_s})$ (respectively $g_2 : x \mapsto f(\frac{x-1}{T_s}) - f(\frac{x}{T_s})$) is strictly increasing (respectively strictly decreasing). Thus the optimal node id replacement that maximizes the KL-divergence gain is obtained by replacing the less frequent node id $v_i$ with the most frequent one $v_j$. □

Algorithm 1 shows an optimal implementation of Lemma 6.2 with respect to the number of performed replacements. This algorithm is run by the adversary. Specifically, the inputs of the algorithm are $\tau_s$ and an input stream $\sigma$ that feeds the sampler component $S_u^s$ of some honest node $u$. Recall that $\tau_s$ is the robustness threshold of the sampling strategy $s$ implemented by $S_u^s$, *i.e.*, for any $\tau_s$-close input stream $\sigma$, the sampling strategy $s$ is able to output a uniform and ergodic sample. The goal of the greedy Algorithm 1 is to tamper with the input stream $\sigma$ in order to increase its KL-divergence above $\tau_s$ with a minimum effort.

**Algorithm 1:** Adversary biasing strategy

```
Data: an input stream σ, the robustness threshold τ_s
Result: the number of replacements ℓ if it exists
1  if τ_s ≥ D^max then
2      return "fail"
3  else
4      ℓ ← 0;
5      v_j ← arg max_{v_j ∈ S} m_{v_j};
6      while (D(q_σ||p^(U)) ≤ τ_s) do
7          v_i ← arg min_{v ∈ S : m_{v_i} ≠ 1} m_{v_i};
8          let k be the index of an item in the part of the
           stream controlled by an adversary such that
           σ[k] = v_i ;
9          σ[k] ← v_j   //one occurrence of v_i is dropped
           and v_j is injected instead;
10         ℓ ← ℓ + 1;
11     return ℓ
```

By assumption, the adversary is omniscient and therefore has the capacity to observe the entire input stream $\sigma$. From Section 4, the adversary knows the strategy $s$ of the sampler, and thus can compute the value of $\tau_s$. The value of the maximum divergence $\mathcal{D}^{\max}$ is computed using Relation (3). If $\mathcal{D}^{\max}$ is larger than or equal to the robustness threshold, the algorithm returns "fail". Otherwise at each iteration, the adversary performs the optimal node id replacement until the KL-divergence exceeds the robustness threshold. Remember however, that the adversary cannot drop messages that have been sent or forwarded by nodes it does not control (i.e. the honest ones). Note that at lines (8) and (9) of Algorithm 1 both $m_{v_i}$ and $m_{v_j}$ are updated. Counter $\ell$ returned by Algorithm 1 represents the number of replacements done by the adversary.

Consider a sampling strategy $s$, its robustness threshold $\tau_s$, and an input stream $\sigma$. Let $\ell$ be the number of replacements executed in Algorithm 1. If we denote by $q_{\sigma(\ell)}$ the probability distribution derived from $\sigma$ after these $\ell$ optimal replacements, then we have

COROLLARY 6.3. (LOWER BOUND ON THE EFFORT EXERTED BY AN OMNISCIENT ADVERSARY)
*The minimum number of replacements an omniscient adversary has to apply to exceed $\tau_s$ is*

$$\delta = \inf\left\{\ell \in \mathbb{N} : \mathcal{D}(q_{\sigma(\ell)}||p^{(\mathcal{U})}) > \tau_s\right\}. \qquad (4)$$

# 7. BLIND ADVERSARY MODEL

In this section, we study the behavior of a blind adversary, that is an adversary that only has the capacity to observe messages sent or received by the nodes he controls. A strategy that the adversary might apply to bias the input stream is to choose a node id (possibly one that belongs to a malicious node but not necessarily) and to push it in the input stream as much as possible. We show that this strategy is optimal with respect to the effort exerted by the adversary and we give the lower bound on the expected minimum amount of work a blind adversary has to exert to bias the input stream.

THEOREM 7.1. (LOWER BOUND ON THE *expected* EFFORT EXERTED BY A BLIND ADVERSARY)
*Let $s$ be a sampling strategy, $\tau_s$ its robustness threshold and $T_s$ the maximum convergence time of $s$. The minimum number of replacements a blind adversary has to apply in expectation to exceed $\tau_s$ is given when the input stream is the uniform one. We have*

$$\tilde{\delta} = \inf\{\ell \in \mathcal{I}_s : \mathcal{R}_\ell > \tau_s\} \tag{5}$$

*where* $\mathcal{I}_s = \left[0..T_s - n + 1 - \left\lfloor \frac{T_s}{n} \right\rfloor\right]$ *and*

$$\mathcal{R}_\ell = \frac{1}{n}\left(\left\lfloor \frac{\ell-1}{\left\lfloor \frac{T_s}{n} - 1\right\rfloor}\right\rfloor \log\left(\frac{T_s}{n}\right) + \log\left(\frac{T_s^2}{T_s + n\ell}\right) - \log\left(T_s - n\left(1 + (\ell-1) \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor\right)\right)\right)$$

PROOF. Let us consider the uniform node ids stream on a window of length $T_s$. For any $v_i \in \mathcal{S}$, $v_i$ is present in the stream $T_s/n$ in average. Thus the probability distribution $p^{(\mathcal{U})}$ is such that $\forall v_i \in \mathcal{S}, p^{(\mathcal{U})}_{v_i} = 1/n$. From Section 6, we have seen that the optimal strategy for the adversary to bias an input stream is to replace the less frequent node id in this stream with the most frequent one. By assumption, the adversary is blind and cannot observe all the node ids of the input stream. Thus the strategy the adversary applies consists in choosing a specific node id $v_j$ and repeatedly pushes $v_j$ in the input stream. Let $\sigma$ be an input stream and $\sigma'$ be the stream obtained from $\sigma$ after one step of this adversarial strategy (*i.e.*, replacing $v_i$ by $v_j$ for some $v_i \in \mathcal{S}$). We have

$$\mathcal{D}(q_{\sigma'}||p^{(\mathcal{U})}) - \mathcal{D}(q_\sigma||p^{(\mathcal{U})}) = \frac{1}{n}\left(\log\left(\frac{m_{v_j}}{m_{v_j}+1}\right) + \log\left(\frac{m_{v_i}}{m_{v_i}-1}\right)\right), \tag{6}$$

where $q_\sigma$ and $q_{\sigma'}$ represent respectively the probability distributions of $\sigma$ and $\sigma'$. In the following, $q_{\sigma(\ell)}$ denotes the probability distribution derived from $\sigma$ after $\ell$ replacements. Given a sampling strategy $s$, we prove by induction on the number of optimal replacements $\ell$ that, starting from a uniform stream, the maximum KL-divergence after $\ell$ replacements is given by $\mathcal{D}(q_{\sigma(\ell)}||p^{(\mathcal{U})}) = \mathcal{R}_\ell$ where

$$\mathcal{R}_\ell = \frac{1}{n}\left(\left\lfloor \frac{\ell-1}{\left\lfloor \frac{T_s}{n} - 1\right\rfloor}\right\rfloor \log\left(\frac{T_s}{n}\right) + \log\left(\frac{T_s^2}{T_s + n\ell}\right) - \log\left(T_s - n\left(1 + (\ell-1) \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor\right)\right)\right) \tag{7}$$

Note that $\ell$ cannot be greater than $\left(T_s - n + 1 - \left\lfloor \frac{T_s}{n} \right\rfloor\right)$. Indeed, all node ids in the initial uniform stream are present at least $\left\lfloor \frac{T_s}{n} \right\rfloor$ times and the maximum number of times a unique id can appear in the stream is $(T_s - n + 1)$.

For $\ell = 1$, the claim immediately holds from Equation 6. Now, assume that the claim also holds for all $1 \leq k \leq \ell$. We show that the claim holds for $k = \ell + 1$. The KL-divergence with respect to the uniform stream after $\ell + 1$ steps is

$$\mathcal{D}(q_{\sigma(\ell+1)}||p^{(\mathcal{U})}) = \mathcal{D}(q_{\sigma(\ell)}||p^{(\mathcal{U})}) + \mathcal{D}(q_{\sigma(\ell+1)}||p^{(\mathcal{U})}) - \mathcal{D}(q_{\sigma(\ell)}||p^{(\mathcal{U})}). \tag{8}$$

The term $\mathcal{D}(q_{\sigma(\ell+1)}||p^{(\mathcal{U})}) - \mathcal{D}(q_{\sigma(\ell)}||p^{(\mathcal{U})})$ represents the gain of step $(\ell+1)$, and $\mathcal{D}(q_{\sigma(\ell)}||p^{(\mathcal{U})})$ is given by Equation 7. Two sub-cases need to be considered: *(i)* $\ell \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor \neq 0$ and *(ii)* $\ell \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor = 0$.
Case $(i)$: the less frequent node id $v_i$ in the stream at step $\ell+1$ is the same as the one removed at step $\ell$. After $\ell$ steps, $m_{v_j} = \frac{T_s}{n} + \ell$ and $m_{v_i} = \frac{T_s}{n} - \left(1 + (\ell-1) \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor\right)$, thus the right part of Equation 6 is equal to

$$\begin{aligned}
&\frac{1}{n}\left(\log\left(\frac{\frac{T_s}{n} + \ell}{\frac{T_s}{n} + \ell + 1}\right) + \log\left(\frac{\frac{T_s}{n} - \left(1 + (\ell-1) \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor\right)}{\frac{T_s}{n} - \left(1 + (\ell-1) \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor\right) - 1}\right)\right) \\
= &\frac{1}{n}\left(\log\left(T_s - n\left(1 + (\ell-1) \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor\right)\right) + \log\left(T_s + n\ell\right) - \log\left(T_s + n(\ell+1)\right) - \log\left(T_s - n\left(2 + (\ell-1) \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor\right)\right)\right).
\end{aligned}$$

By assumption *(i)*, we have $\left\lfloor \frac{\ell-1}{\left\lfloor \frac{T_s}{n} - 1\right\rfloor}\right\rfloor = \left\lfloor \frac{\ell}{\left\lfloor \frac{T_s}{n} - 1\right\rfloor}\right\rfloor$ and

$$\left(1 + (\ell-1) \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor\right) = \left(\ell \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor\right).$$

From Equation 8, we get $\mathcal{D}(q_{\sigma(\ell+1)}||p^{(\mathcal{U})}) =$

$$= \frac{1}{n}\left(\left\lfloor \frac{\ell}{\left\lfloor \frac{T_s}{n} - 1\right\rfloor}\right\rfloor \log\left(\frac{T_s}{n}\right) + \log\left(\frac{T_s^2}{T_s + n(\ell+1)}\right) - \log\left(T_s - n\left(1 + \ell \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor\right)\right)\right),$$

which ends Case $(i)$.
Case $(ii)$. The argumentation is the same as above. However, as $\ell \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor = 0$, the node id that has been previously replaced is now present exactly once in the stream. Thus the adversary needs to randomly choose another node id in the stream before processing the next step of his strategy. Thus applying Equation 6 at step $\ell + 1$ gives

$$\mathcal{D}(q_{\sigma(\ell+1)}||p^{(\mathcal{U})}) - \mathcal{D}(q_{\sigma(\ell)}||p^{(\mathcal{U})}) = \frac{1}{n}\left(\log\left(\frac{\frac{T_s}{n} + \ell}{\frac{T_s}{n} + \ell + 1}\right) + \log\left(\frac{\frac{T_s}{n}}{\frac{T_s}{n} - 1}\right)\right). \tag{9}$$

By assumption $\left((\ell-1) \bmod \left\lfloor \frac{T_s}{n} - 1\right\rfloor = \left\lfloor \frac{T_s}{n} - 1\right\rfloor - 1\right)$, and by combining the induction hypothesis 7 with the gain obtained at step $\ell + 1$ (Equation 9), we get $\mathcal{D}(q_{\sigma(\ell+1)}||p^{(\mathcal{U})}) =$

$$\frac{1}{n}\left(\left\lfloor \frac{\ell-1}{\left\lfloor \frac{T_s}{n} - 1\right\rfloor}\right\rfloor \log\left(\frac{T_s}{n}\right) + 3\log\left(T_s\right) - \log\left(T_s + n(\ell+1)\right) - \log\left(T_s - n\right) - \log\left(n\right)\right).$$

By assumption of the case : $\left\lfloor \frac{\ell}{\left\lfloor \frac{T_s}{n} - 1\right\rfloor}\right\rfloor = \left\lfloor \frac{\ell-1}{\left\lfloor \frac{T_s}{n} - 1\right\rfloor}\right\rfloor + 1$, which proves the induction: $\mathcal{D}(q_{\sigma(\ell+1)}||p^{(\mathcal{U})}) =$

$$= \frac{1}{n}\left(\left\lfloor \frac{\ell}{\left\lfloor \frac{T_s}{n} - 1\right\rfloor}\right\rfloor \log\left(\frac{T_s}{n}\right) + \log\left(\frac{T_s^2}{T_s + n(\ell+1)}\right)\right.$$

$$-\log\left(T_s - n\left(1+\ell \mod \left\lfloor \frac{T_s}{n} - 1 \right\rfloor\right)\right)\right).$$

As a conclusion, any value of $\ell$ that allows the adversary to exceed the robustness threshold $\tau_s$ defeats the sampling strategy. Thus, the minimum number of replacement operations $\tilde{\delta}$ is the lower bound of this set of values. □

We now evaluate the minimum amount of work a blind adversary has to exert, in the worst case, to bias the input stream. In the worst case, the node id $v_i$ the adversary has chosen to blindly flood might be initially present only once in the input stream. In order to bias the input stream, the adversary needs to push id $v_i$ sufficiently often so that the probability of appearance of id $v_i$ reaches the uniform value, with respect to all the other node ids, and then to continue to push this id $\tilde{\delta}$ times so that the divergence between the resulting stream and the uniform one is maximum.

THEOREM 7.2. (LOWER BOUND ON THE EFFORT EXERTED BY A BLIND ADVERSARY)
*Let $s$ be a sampling strategy, $\tau_s$ its robustness threshold and $T_s$ the maximum convergence time of $s$. The minimum number of replacements the adversary has to apply on a stream, in the worst case, to exceed $\tau_s$ is*

$$\tilde{\delta} + \left\lceil \frac{T_s}{n} \right\rceil - 1.$$

PROOF. The proof is immediate. First, the adversary has to raise the chosen id at least up to the uniform value. As in the worst case, this id is present only once in the initial stream, this costs $\lceil \frac{T_s}{n} \rceil - 1$ replacements to reach a number of occurrences equals to $\lceil \frac{T_s}{n} \rceil$. Moreover, once this id is present in the modified stream $\lceil \frac{T_s}{n} \rceil$ times, the adversary follows the same strategy as before, which requires $\tilde{\delta}$ more steps to guarantee that the robustness threshold $\tau_s$ is exceeded. Note that this value is a worst-case bound and not the exact minimum value with respect to $\tau_s$ because after the first $(\lceil \frac{T_s}{n} \rceil - 1)$ steps, the modified stream could be different from the uniform one. In this situation, the KL-divergence to the uniform stream is strictly greater than 0, reducing accordingly the amount of work of the adversary to exceed $\tau_s$. □

# 8. CONCLUSION AND OPEN ISSUES

In this paper, we have focused on the problem of achieving uniform and ergodic peer sampling in large scale open systems potentially populated with malicious peers. This problem consists in guaranteeing that the knowledge of the system maintained by each honest peer is a uniform and non permanent sample of the whole population of the system. By modeling input streams as probability distributions, we have characterized the minimum effort (measured in terms of node ids replacements) that an omniscient and a blind adversary have to exert on the input stream of node identifiers to exceed the robustness threshold that quantifies the power of a sampling strategy. Similarly to (pseudo)-random number generators that are considered as the basic mathematical tool to generate complex probability distributions, we believe that uniform peer sampling should be regarded as a necessary building block to derive larger classes of sampling schemes. This building block is of utmost importance in systems in which the population is continuously evolving and thus, where it is impossible to capture the full complexity of the network through global snapshots.

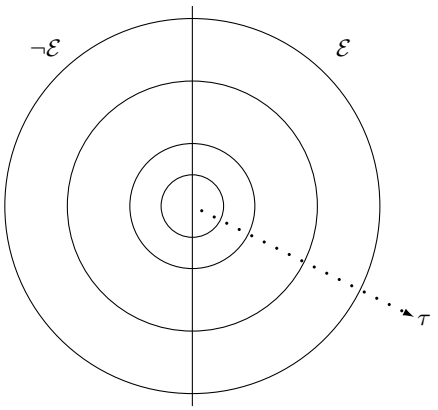


**Figure 2: Characterization of Sampling Strategies**

We conjecture that there exists a total order relationship on the power of (non-)ergodic uniform sampling strategies with respect to their robustness threshold $\tau$. This can be depicted by using a planar representation as shown by Figure 2. Each $\tau$-radius circle in this picture represents the class of $\tau$-close uniform sampling strategies. For instance, in this representation the strategy proposed by Bortnikov *et al.* [7] belongs to the largest circle corresponding to $\tau = \infty$ on its non ergodic part (left side). As another illustrative example, the sampling strategy proposed in Busnel *et al.* [8] should be ranked as less powerful than the one proposed by Gurevich *et al.* [11] since the latter one achieves uniform and ergodic sampling despite message loss contrary to the former one. Both strategies would appear in the ergodic part of the representation (right side). Finally, we think that this classification can be used as a tool to precisely compare any two sampling strategies as different as they are (*e.g.*, [4] and [19]). As future work, we intend to rank the state-the-art sampling algorithms in the light of this new framework. Moreover, we plan to extend this work by integrating other dimensions, such as space and time resources, and deterministic *versus* probabilistic strategies.

# 9. REFERENCES


[1] S. M. Ali and S. D. Silvey. General Class of Coefficients of Divergence of One Distribution from Another. *Journal of the Royal Statistical Society. Series B (Methodological)*, 28(1):131–142, 1966.

[2] E. Anceaume, F. V. Brasileiro, R. Ludinard, B. Sericola, and F. Tronel. Dependability Evaluation of Cluster-based Distributed Systems. *International Journal of Foundations of Computer Science (IJFCS)*, 5(22), 2011.

[3] E. Anceaume, Y. Busnel, and S. Gambs. Uniform and Ergodic Sampling in Unstructured Peer-to-Peer Systems with Malicious Nodes. In *Proceedings of the 14th International Conference On Principles Of Distributed Systems (OPODIS)*, volume 6490, pages 64–78, 2010.

[4] B. Awerbuch and C. Scheideler. Towards a Scalable and Robust Overlay Network. In *Proceedings of the 6th International Workshop on Peer-to-Peer Systems (IPTPS)*, 2007.

[5] R. Baldoni, R. Beraldi, V. Quema, L. Querzoni, and S. Tucci-Piergiovanni. TERA: Topic-based Event Routing for Peer-to-peer Architectures. In *Proceedings of the International Conference on Distributed Event-Based Systems (DEBS)*, pages 2–13. ACM, 2007.

[6] M. Bertier, Y. Busnel, and A.-M. Kermarrec. On Gossip and Populations. In *Proceedings of the 16th International Colloquium on Structural Information and Communication Complexity (SIROCCO)*, 2009.

[7] E. Bortnikov, M. Gurevich, I. Keidar, G. Kliot, and A. Shraer. Brahms: Byzantine Resilient Random Membership Sampling. *Computer Networks*, 53:2340–2359, 2009. A former version appeared in the 27th ACM Symposium on Principles of Distributed Computing (PODC), 2008.

[8] Y. Busnel, R. Beraldi, and R. Baldoni. On the Uniformity of Peer Sampling based on View Shuffling. *Journal of Parallel and Distributed Computing*, To appear (In Press, Corrected Proof), 2011.

[9] T. Cover and J. Thomas. Elements of information theory. *Wiley New York*, 1991.

[10] P. B. Godfrey, S. Shenker, and I. Stoica. Minimizing churn in distributed systems. In *Proceedings of the ACM SIGCOMM*, 2006.

[11] M. Gurevich and I. Keidar. Correctness of Gossip-Based Membership under Message Loss. In *Proceedings of the 28th annual Symposium on Principles of distributed computing (PODC)*, Calgary, AL, Canada, 2009. ACM Press.

[12] M. Jelasity, S. Voulgaris, R. Guerraoui, A.-M. Kermarrec, and M. van Steen. Gossip-based Peer Sampling. *ACM Transaction on Computer System*, 25(3), 2007.

[13] G. P. Jesi, A. Montresor, and M. van Steen. Secure Peer Sampling. *Computer Networks*, 54(12):2086–2098, 2010.

[14] D. R. Karger and M. Ruhl. Simple Efficient Load Balancing Algorithms for Peer-to-Peer. In *Proceedings of the 3rd International Workshop on Peer-to-Peer Systems (IPTPS)*, 2004.

[15] S. Le Blond, F. Le Fessant, and E. Le Merrer. Finding Good Partners in Availability-Aware P2P Networks. In *Proceedings of the 11th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS)*, pages 472–484. Springer-Verlag, 2009.

[16] D. Liu, P. Ning, and W. Du. Detecting Malicious Beacon Nodes for Secure Location Discovery in Wireless Sensor Networks. In *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2005.

[17] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker. Search and Replication in Unstructured Peer-to-Peer Networks. In *Proceedings of the International Conference on Supercomputing (ICS)*, pages 84–95, 2002.

[18] D. Malkhi, M. K. Reiter, A. Wool, and R. N. Wright. Probabilistic Byzantine Quorum Systems. In *Proceedings of the 21rst annual ACM Symposium on Principles of Distributed Computing (PODC)*, 2008.

[19] L. Massoulié, E. L. Merrer, A.-M. Kermarrec, and A. Ganesh. Peer Counting and Sampling in Overlay Networks: Random Walk Methods. In *Proceedings of the 25th Annual Symposium on Principles of Distributed Computing (PODC)*, pages 123–132. ACM Press, 2006.

[20] G. Monge. Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Academie royale des sciences, avec les Mémoires de Mathématique et de Physique*, pages 666–704, 1781.

[21] A. Singh, T.-W. Ngan, P. Druschel, and D. S. Wallach. Eclipse Attacks on Overlay Networks: Threats and Defenses. In *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM)*, 2006.

[22] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao. SybilLimit: A Near-Optimal Social Network Defense against Sybil Attacks. In *Proceedings of the IEEE Symposium on Security and Privacy (SP)*, pages 3–17, 2008.

[23] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman. SybilGuard: Defending against Sybil Attacks via Social Networks. In *Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, pages 267–278, 2006.